Automatic Labeling Hierarchical Topics

Xian-Ling Mao[♠†], Zhao-Yan Ming[♡], Zheng-Jun Zha[♡], Tat-Seng Chua[♡], Hongfei Yan[♠‡], Xiaoming Li[♠]
[♠]Department of Computer Science and Technology, Peking University, China
[♡]School of Computing, National University of Singapore, Singapore
{xianlingmao, yanhf, lxm}@pku.edu.cn, {chuats,mingzy, zhazj}@comp.nus.edu.sg

ABSTRACT

Recently, statistical topic modeling has been widely applied in text mining and knowledge management due to its powerful ability. A topic, as a probability distribution over words, is usually difficult to be understood. A common, major challenge in applying such topic models to other knowledge management problem is to accurately interpret the meaning of each topic. Topic labeling, as a major interpreting method, has attracted significant attention recently. However, previous works simply treat topics individually without considering the hierarchical relation among topics, and less attention has been paid to creating a good hierarchical topic descriptors for a hierarchy of topics. In this paper, we propose two effective algorithms that automatically assign concise labels to each topic in a hierarchy by exploiting sibling and parent-child relations among topics. The experimental results show that the inter-topic relation is effective in boosting topic labeling accuracy and the proposed algorithms can generate meaningful topic labels that are useful for interpreting the hierarchical topics.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Text Mining

Keywords

Statistical topic models, topic model labeling

1. INTRODUCTION

Statistical topic modeling has been widely applied in text mining and knowledge management due to its broad applications, such as word sense disambiguation [2, 3], temporal analysis [17, 18] and opinion mining [10, 7] etc.

A wealth of topic models have been proposed to extract interesting topics in the form of multinomial distributions from the corpus automatically. A common, major challenge in applying all such topic models is to accurately interpret the meaning of each topic. In general, it is very difficult for users to understand a topic merely based on the multinomial word distribution, especially when they are not familiar with the source collection. Topic labeling, which generates meaningful labels for a topic so as to facilitate topic interpretation, has attracted increasing attention recently. An example of a hierarchy of topics and their labels are showed in Fig 1.

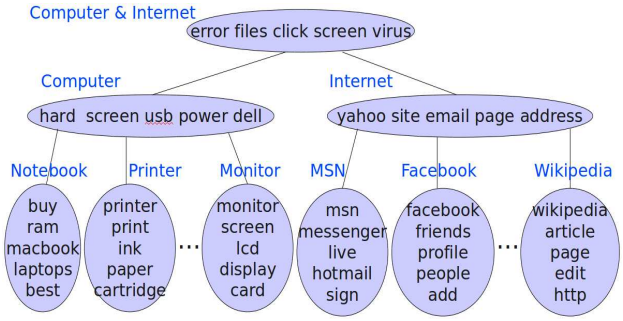

Figure 1: A topical hierarchy and lables; The top 5 words are shown for each topic.

Early research on topic labeling generally either select top words in the distribution as primitive labels [1, 14], or generate labels manually in a subjective manner [10, 12, 15]. However, it is highly desirable to automatically generate meaningful labels. Several automatical labeling methods have been proposed recently. Mei et al. (2007) [11] proposed to automatically label multinomial topic models by casting this labeling problem as an optimization problem. Magatti et al. (2009) [8] proposed an algorithm for labeling topics automatically according to a given topic hierarchy. Lau et al. (2010) [6] labeled topics via selecting one of the top-10 topic terms to label the overall topics. Lau et al. (2011) [5] reused the selection of features proposed by Lau et al. (2010) [6], and also enlarged the candidate labels by using Wikipedia. Despite the success of these works, they simply treat topics individually without considering the hierarchical relation among topics. The descriptors for hierarchical topics generated by these methods are often incomprehensible and are not consistent with the hierarchical relation among topics.

In this paper, we propose an automatic topic labeling ap-

[†] This work was done in National University of Singapore.
[‡] Corresponding author.

CIKM'12, October 29–November 2, 2012, Maui, HI, USA.

proach, which improves the labeling accuracy by exploiting the sibling and parent-child relations among topics.

2. NOTATIONS

Formally, we define D as the set of all documents. The hierarchical structure between topics is formalized as *is-a* relationship $t_j \rightarrow t_i$, defining that t_j is the parent (direct parent) of t_i. For specifying the set of topics with direct parent t_j, we use $T_{t_j \rightarrow *}$ to denote all the documents contained in this sub-hierarchy as $D_{t_j \rightarrow *}$. For a topic t_j, we use $T_{t_j \Rightarrow *}$ to denote the set of all its direct children and grand-children and so forth. $D_{t_j \Rightarrow *}$ denotes all the documents contained in hierarchy $T_{t_j \Rightarrow *}$. Semantically, the *is-a* relationship assumes that if a document is assigned to a topic, it is also assigned to its parent topic.

3. HIERARCHICAL TOPICS LABELING

After assigning each document to its top topic according to the topic distribution on this document, two steps are executed to generate meaningful labels for topics: (1) extract candidate labels; (2) rank candidate labels for each topic.

3.1 Candidate Label Extraction

As discussed in Mei et al. (2007) [11], compared with single terms and sentences, phrases appear to be more appropriate for labeling a topic. In general, to obtain phrases in documents, there are two basic approaches: *Chunking Parsing* and *Ngram Testing* [11, 4]. The *Ngram Testing* usually performs better than *Chunking Parsing* in labeling topics [11]. Therefore, we first generate meaningful phrases as candidate labels using *Ngram Testing* [4]. In addition, it is also possible to label a topic better using its own topic terms, as demonstrated by [6]. Thus we also add the top-n topic terms into the set of candidate labels, based on the term probabilities on the corresponding topic.

3.2 Structure Assisted Label Ranking

We rank the candidate labels, by exploiting the structural relation among the topics. To incorporate structural relation, we have the following four intuitive assumptions: **(A1)** the label of a topic should be representative and important terms in this topic. **(A2)** given a topic, terms that are more common among its child topics are more likely to be suitable labels; For example, assume that the term "classification" occurs in documents of the topics about "support vector machine" and "naive bayes"; Whereas terms "support vector machine" and "naive bayes" only occur in their respective documents of each topic. It's reasonable to label term "classification" to the parent topic of the topics about SVM and NB. **(A3)** the labels near the top of the hierarchy should be more general than those at the bottom; For example, the terms "support vector machine" is more specialized than "machine learning". **(A4)** if one label occurs often in one sibling topic only, then this label should be preferred over labels that occurr in all sibling topics. The assumptions **A2** and **A3** are about parent-child relations; and **A4** is about sibling relations. However, it is difficult to weight the structual relation. Based on these four assumations, we will describe the structual relation by *the term weight scoring* and *statistic scoring* approaches.

3.2.1 Term Weighting Based Ranking

The use of global term weighting schemes like TFIDF or Okapi BM25 [9] helps to improve the discrimination capability of labels based on the underlying document distribution. Given a topic t_j and a candidate label c_i, to describe the assumption A1, we use following scoring function:

$$\mathcal{S}_1(t_j, c_i) = idf(D, c_i) \cdot \sum_{d_k \in D_{t_j \rightarrow *}} tf(d_k, c_i) \quad (1)$$

where $idf(D, c_i) = \log(\frac{|D|}{\#(D, c_i)} + 1)$ with $\#(D, c_i)$ equals to the number of documents in the collection D containing term c_i, and $tf(d_k, c_i) = \#(d_k, c_i)$ where $\#(d_k, c_i)$ is the number of term c_i in document d_k. To describe the assumption A4, we use following two scoring function:

$$\mathcal{S}_2(t_j, c_i) = idf(D_{t_p \rightarrow *}, c_i) \quad (2)$$

where $idf(D_{t_p \rightarrow *}, c_i)$ is the inverse document frequency vector over the document collection $D_{t_p \rightarrow *}$ with $t_p \rightarrow t_j$ and t_p is defined as the parent topic of t_j.

$$\mathcal{S}_3(t_j, c_i) = \log(\frac{\#(t_p)}{\#(c_i, t_p)} + 1) \cdot \exp(\frac{\#(c_i, D_{t_j \rightarrow *})}{|D_{t_j \rightarrow *}|}) \quad (3)$$

where $\#(c_i, t_p)$ being the number of direct child topics of t_p containing term c_i and $\#(t_p)$ being the number of direct child topics.

By denoting the path length between two topics as $l(j, i)$, to describe the assumption A2 and A3, we use following scoring function:

$$\mathcal{S}_{tw}(t_j, c_i) = \sum_{t_k \in T_{t_j \Rightarrow *}} \frac{f(T_{t_j \rightarrow *}, c_i) \cdot stat(t_j, t_k, c_i)}{l(j, k)} \quad (4)$$

where $f(T_{t_j \rightarrow *}, c_i)$ is the topic frequency of term (c_i), based on the sibling topics $(T_{t_j \rightarrow *})$ of topic t_j. When $stat(t_j, t_k, c_i) = \mathcal{S}_1(t_k, c_i) \cdot \mathcal{S}_2(t_j, c_i) \cdot \mathcal{S}_3(t_j, c_i)$, $\mathcal{S}_{tw}(t_j, c_i)$ incorporates all assumptions, we refer to this method as *TWL*.

3.2.2 Statistical Significance Based Ranking

Comparative statistics like Jensen-Shannon Divergence (JSD) are able to estimate with statistical signicance whether the occurrences of a term differ between the documents assigned with a topic and a reference collection. Such terms yield good labels for a topic.

To describe the assumption A2, we define the reference collection with respect to topic t_j as all documents assigned to the child topics of its direct parent excluding all documents contained in t_j. Formally, the scoring function is:

$$\mathcal{R}_1(t_j, c_i) = P(c_i) \log \frac{P(c_i)}{M(c_i)} + Q(c_i) \log \frac{Q(c_i)}{M(c_i)} \quad (5)$$

with $\mathcal{P}(c_i) = \frac{\#(c_i, D_{t_p \rightarrow * \setminus t_j})}{\sum_{w' \in W} \#(w', D_{t_p \rightarrow * \setminus t_j})}$, $\mathcal{Q}(c_i) = \frac{\#(c_i, D_{t_j \rightarrow *})}{|D_{t_j \rightarrow *}|}$ and $\mathcal{M}(c_i) = \frac{1}{2}(P(c_i) + Q(c_i))$.

Similar to the TWL approach, to describe assumption A2 and A3, we can use Formula (4). Thus, we modify the *stat* part of in Formula (4) as:

$$stat(t_j, t_k, c_i) = \mathcal{S}_1(t_k, c_i) \cdot \mathcal{R}_1(t_j, c_i) \quad (6)$$

Then, we obtain the final scoring formula as follows:

$$\mathcal{R}_{ss}(t_j, c_i) = \sum_{t_k \in T_{t_j \Rightarrow *}} \frac{f(T_{t_j \rightarrow *}, c_i) \cdot \mathcal{S}_1(t_k, c_i) \cdot \mathcal{R}_1(t_j, c_i)}{l(j, k)} \quad (7)$$

and we refer to this method as *JSD*.

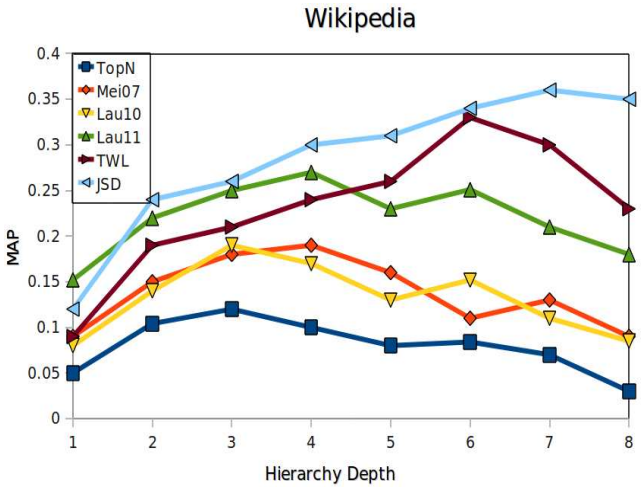

Figure 2: Labeling performance on the Wikipedia dataset by different approaches with various hierarchy depth.

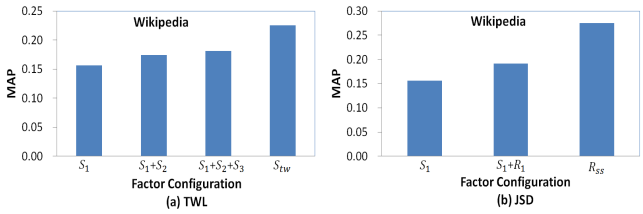

Figure 3: Labeling performance with various factor configuration over the Wikipedia dataset. S_i and R_i are the factors defined in Section 3.2, and "+" means combination; S_{tw} and R_{ss} are *TWL* and *JSD* methods.

4. EXPERIMENTS

4.1 Data

We first crawled question-answer pairs (QA pairs) from two top categories in Yahoo! Answers: *Computers & Internet* and *Health*. This gives rise to an archive of 6,345,786 QA documents. We refer to the dataset as ***Y!A***. Moreover, we crawled documents from the Wikipedia, and filtered out categories that do not carry any semantic information. In order to create a tree structure, we chose four topics as starting point, i.e. *arts, computing, health, and sports*, and then traversed the graph in a breath-first manner with a maximum depth of 10. For each node, we randomly chose only 5 outgoing links and 80 documents. Thus, we obtained about 1,640 categories and about 20,572 documents as the test dataset. We refer to the dataset as ***Wiki***.

To obtain groundtruth, i.e. a hierarchy of topics and corresponding labels, we conducted the following steps: (1) create preliminary topic hierarchies and labels by using a supervised hierarchical topic model, i.e. *hierarchical Labeled LDA* (hLLDA) [13], over our document collections. (2) judge the resultant hierarchies manually, and correct inaccurate labels.

4.2 Experimental Setting

For evaluating the correctness of the generated topic labels, we used the following two definitions to judge what is a correct label: **exact match** and **partial match** [16]. For a given topic with correct-label C and its parent-label P, a label L is an *exact match* of the correct label C if there exists a synonym SL of L such that SL is one of "C", "C P" and "P C." On the other hand, a label L is a *partial match* of the correct label C if there exists a synonym CL of L such that CL has at least one term same as that in "C", "C P" and "P C."

For each of the two definitions of a correct label, we compute the following performance metric: (1) *Match at top N results (**Match@N**)*, which indicates whether the top N results contain any correct labels; (2) *Precision at top N results (**P@N**)*; (3) *Mean average precision (**MAP**)*; and (4) *Mean Reciprocal Rank (**MRR**)*.

We compared the proposed TWL and JSD methods with the following four state-of-the-art methods: (1) **TopN**: Labeling a topic using the top k terms in its word distribution [1]; (2) **Mei07**: proposed by Mei et al. (2007) [11]; (3) **Lau10**: proposed by Lau et al. (2010) [6]; and, (4) **Lau11**: proposed by Lau et al. (2011) [5].

Table 1: Match@N, Precision@N and MRR with the exact and partial match criteria. "(E)" means "exact match"; and "(P)" means "partial match".

	Method	Match@N			P@N			MRR
		N=1	N=3	N=5	N=1	N=3	N=5	
Y!A (E)	TopN	0.271	0.313	0.354	0.271	0.125	0.092	0.304
	Mei07	0.236	0.288	0.327	0.236	0.186	0.128	0.273
	Lau10	0.167	0.229	0.292	0.167	0.104	0.088	0.224
	Lau11	**0.283**	0.328	0.362	0.283	0.203	0.183	0.319
	TWL	0.273	**0.348**	**0.393**	**0.273**	**0.239**	**0.204**	**0.335**
	JSD	0.252	**0.376**	**0.448**	**0.252**	**0.244**	**0.216**	**0.373**
Y!A (P)	TopN	0.438	0.583	0.771	0.438	0.257	0.221	0.551
	Mei07	0.453	0.635	0.790	0.453	0.363	0.282	0.617
	Lau10	0.292	0.458	0.625	0.292	0.208	0.192	0.431
	Lau11	0.342	0.493	0.678	0.342	0.268	0.212	0.487
	TWL	**0.494**	**0.654**	**0.811**	**0.494**	**0.395**	**0.325**	**0.631**
	JSD	**0.525**	**0.692**	**0.867**	**0.525**	**0.458**	**0.336**	**0.658**
Wiki (E)	TopN	0.104	0.188	0.223	0.104	0.079	0.062	0.175
	Mei07	0.189	0.204	0.232	0.189	0.155	0.131	0.199
	Lau10	0.143	0.194	0.249	0.143	0.118	0.084	0.192
	Lau11	0.251	0.272	0.326	0.251	0.215	0.180	0.278
	TWL	**0.355**	**0.373**	**0.392**	**0.355**	**0.319**	**0.283**	**0.356**
	JSD	**0.373**	**0.407**	**0.448**	**0.373**	**0.342**	**0.290**	**0.390**
Wiki (P)	TopN	0.172	0.266	0.423	0.172	0.113	0.083	0.254
	Mei07	0.284	0.417	0.543	0.284	0.160	0.112	0.403
	Lau10	0.194	0.358	0.446	0.194	0.143	0.099	0.322
	Lau11	0.312	0.538	0.621	0.312	0.189	0.103	0.480
	TWL	**0.360**	**0.552**	**0.648**	**0.360**	**0.219**	**0.138**	**0.520**
	JSD	**0.396**	**0.589**	**0.673**	**0.396**	**0.235**	**0.151**	**0.546**

4.3 Results

Table 1 presents the performance comparison among the six methods over the two datasets. For Yahoo! Answers, the Match@1 value is around 0.25 in exact match for JSD, and this value is lower than that of TopN and Lau2010. This unexpected result may be caused by the heavy noise information in Yahoo! Answers. However, our approaches outperform the other approaches in terms of all the other measure values over the two datasets. By t-test with 95% significance, there is significant difference between our proposed methods and the others. The above results demonstrate that the capacity of the hierarchical relation among topics in facilitating topic labeling.

Moreover, we can see that JSD outperforms TWL. This may be because JSD uses probabilistic distributions to describe sibling and parent-child relations, and it is more effective than the term weighting method used in TWL.

4.4 Effectiveness of Hierarchy Depth

To explore the influence of the hierarchical depth on la-

beling performance, we executed the labeling methods with the utilization of varying hierarchy depth from level 1 to 8. Figure 2 shows the labeling accuracies with respect to different hierarchy depth on the Wikipedia dataset. We can see that the labeling accuracy varies dramatically with the increase of hierarchy depth; and our proposed methods signicantly outperform the other methods that do not exploit hierarchical relation at most cases.

4.5 Effectiveness of the Factors

We have proposed some factors to measure our assumptions in Section 3.2, such as S_i and R_i. It is necessary to evaluate the influence of each factor on the labeling performance. Figure 3 shows the labeling accuracies by the TWL and JSD methods with different factor combination over the Wikipedia dataset. The labelling performance increases with the incorporation of more hierarchical relation, i.e., the utilization of more factors. Specifically, the performance increases when one more factor comes in. For example, the performance of the combination of S_1 and S_2 is better than that of S_1. These results demonstrate the reasonability of our assumptions in Section 3.2.

5. CONCLUSION AND FUTURE WORK

In this paper, we have proposed two automatic methods for labeling topics in a hierarchy. Experiments over two real-word document corpus shown that exploiting the hierarchical relation among topics yield a statistically significant performance improvements as compared to the state-of-the-arts that simply treat the topics individually.

The future work is to incorporate domain ontologies or existing taxonomies, such as Wordnet and Wikipedia, into the topic labeling process.

Acknowledgments

This work was partially supported by NSFC with Grant No. 61073082, 60933004, 70903008 and NExT Search Centre, which is supported by the Singapore National Research Foundation & Interactive Digital Media R&D Program Office, MDA under research grant (WBS:R-252-300-001-490).

6. REFERENCES

[1] D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022, 2003.

[2] J. Boyd-Graber and D. Blei. Syntactic topic models. *Arxiv preprint arXiv:1002.4665*, 2010.

[3] J. Boyd-Graber, D. Blei, and X. Zhu. A topic model for word sense disambiguation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 1024–1033, 2007.

[4] J. Chen, J. Yan, B. Zhang, Q. Yang, and Z. Chen. Diverse topic phrase extraction through latent semantic analysis. In *Data Mining, 2006. ICDM'06. Sixth International Conference on*, pages 834–838. IEEE, 2006.

[5] J. Lau, K. Grieser, D. Newman, and T. Baldwin. Automatic labelling of topic models. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 1536–1545. Association for Computational Linguistics, 2011.

[6] J. Lau, D. Newman, S. Karimi, and T. Baldwin. Best topic word selection for topic labelling. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 605–613. Association for Computational Linguistics, 2010.

[7] Y. Lu and C. Zhai. Opinion integration through semi-supervised topic modeling. In *Proceeding of the 17th international conference on World Wide Web*, pages 121–130. ACM, 2008.

[8] D. Magatti, S. Calegari, D. Ciucci, and F. Stella. Automatic labeling of topics. In *Intelligent Systems Design and Applications, 2009. ISDA'09. Ninth International Conference on*, pages 1227–1232. IEEE, 2009.

[9] C. Manning, P. Raghavan, and H. Schutze. *Introduction to information retrieval*, volume 1. Cambridge University Press Cambridge, 2008.

[10] Q. Mei, C. Liu, H. Su, and C. Zhai. A probabilistic approach to spatiotemporal theme pattern mining on weblogs. In *Proceedings of the 15th international conference on World Wide Web*, pages 533–542. ACM, 2006.

[11] Q. Mei, X. Shen, and C. Zhai. Automatic labeling of multinomial topic models. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 490–499. ACM, 2007.

[12] Q. Mei and C. Zhai. Discovering evolutionary theme patterns from text: an exploration of temporal text mining. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 198–207. ACM, 2005.

[13] Y. Petinot, K. McKeown, and K. Thadani. A hierarchical model of web summaries. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 670–675. Association for Computational Linguistics, 2011.

[14] D. Ramage, P. Heymann, C. Manning, and H. Garcia-Molina. Clustering the tagged web. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, pages 54–63. ACM, 2009.

[15] Y. Teh, M. Jordan, M. Beal, and D. Blei. Hierarchical dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2006.

[16] P. Treeratpituk and J. Callan. Automatically labeling hierarchical clusters. In *Proceedings of the 2006 international conference on Digital government research*, pages 167–176. ACM, 2006.

[17] C. Wang, D. Blei, and D. Heckerman. Continuous time dynamic topic models. In *Proc. of UAI*. Citeseer, 2008.

[18] X. Wang and A. McCallum. Topics over time: a non-markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 424–433. ACM, 2006.